Is blockchain a trustworthy technology?

Maryline Laurent

Laurent, M.

«Is blockchain a trustworthy technology?»

in **Signs of trust – The impact of seals on personal data management** (Chapter 11, pages 180 to 198). Coordinated by Claire Levallois-Barth, Chair Values and Policies of Personal Information (France), January 2018.

Handbook available in electronic version: http://www.personal-information.org/
Also available in paper format: ISBN 978-2-9557308-6-7

Is blockchain a trustworthy technology?

This chapter relies on a specific example of technology, the blockchain, to explain how the concept of “trust by design” presented in Chapter 1 can be implemented and what its limits are.

Blockchain technology was developed towards the end of the 2000s, within a wider project related to the transfer of cryptocurrencies over the Internet: Bitcoin. This project made blockchain technology popular and demonstrated its reliability. In 2014, the not-for-profit Ethereum headed by Vitalik Buterin began working on the idea that this technology should be extended to include some code to enable a new type of transaction: “smart contracts.”

- Examples of smart contracts include launching a cryptocurrency transfer once a parcel is delivered or prepaying for a rental service in order to open a door (e.g. of a vehicle or a house).

In 2015, a first version of the source code of Ethereum was made public, allowing many industrial players and independent developers to innovate and offer services on top of this technology. Recently, Axa issued Fizzy, which offers compensation to passengers whose flights are delayed.[1]

1 https://www.coindesk.com/axa-using-ethereums-blockchain-new-flight-insurance-product

Blockchain is often compared to a large, publicly accessible and auditable ledger managed by its "members." Members can add entries to the ledger after obtaining approval from several other members, or in some cases the majority. It is therefore possible to track the entries added by each member, without necessarily knowing who wrote the entries since members use pseudonyms.[2]

After introducing the fundamental building blocks that help understand blockchain (11.1), we describe how this technology works at a technical level (11.2.). We then identify the key features that introduce a level of trust (11.3.). Finally, we draw up an overview of the risks and limits associated with this technology and discuss its ability to guarantee personal data protection (11.4. and 11.5.).

A major difficulty is to tease out the features strictly associated with the concept of blockchain and the ones associated with its different implementations, e.g. Bitcoin, Ethereum, Ripple, or Litecoin.[3] The explanations we give in this article are mostly related to Bitcoin, which is more consistently studied in the literature.

2 A member is pseudonymous when they are using an alias instead of their actual identity.

3 We will refer to these specific implementations as Bitcoin blockchain, Ethereum blockchain, and so on.

11.1. The fundamental building blocks

The security that blockchain offers mostly relies on standard cryptographic mechanisms, notably public key cryptography, hash functions and digital signatures.

Cryptographic mechanisms

Public key cryptography implies that every entity in a system has two keys: a public key shared with everyone, and a private key known only to the owner. The ***private key*** is a binary string enabling owners to prove their identities, e.g. sign a transaction request to prove they initiated it. The public key allows other entities to authenticate this signature.

The security level of a cryptosystem can be measured by how hard it is to find its private keys. This level is directly proportional to the size of the parameters: the larger they are, the harder it is to find the private key. However, as computers become cheaper and their processing power and memory bigger, the size of these parameters needs to increase on a regular basis in order to maintain the same security level. This level is measured by how many operations the attacker needs to make in order to crack the cryptosystem. Nowadays, a security level of 100 is considered sufficient — meaning attackers would need to perform 2^{100} operations to break the system.

The Bitcoin project relies on Elliptic Curve Cryptography (ECC) and the Elliptic Curve Digital Signature Algorithm (ECDSA). ECDSA uses elliptic curves to provide keys that are reasonably sized compared to other public key infrastructures such as RSA for the same level of security. For instance, in RSA (named after its inventors, Rivest, Shamir and Adleman), for a security level of 112 (2^{112} operations), the RSA key length is 3072 bits (i.e. a string of 3072 zeros and ones) while the ECC key length is only 256 bits.

Hash functions

Hash functions are very important in blockchains, especially SHA256. They allow to use private keys to craft signatures and authenticate transactions, reliably link a blockchain member to their public key, and therefore identify the source of a transaction or a block into the blockchain. They are also used to create chained links between the blocks so that their order cannot be modified, therefore offering some form of guarantee of the blockchain's integrity.

What are the properties of cryptographic hash functions

- They give a fixed-size result (or a *hash*): regardless of the entry, the function always returns a result of the same size. For instance, SHA256 always returns a fixed 256-bit hash.
- One-way function: it is very difficult[4] to find the entry based only on the function's result;
- Collision resistance: it is very difficult to obtain the same result for two different entries;
- Avalanche effect: changing one bit in the entry entails a change in more than half of the result's bits. This property is crucial in guaranteeing the integrity of entries since any modification is easily detected.

Digital signatures

The signing process begins with the application of a hash function to the elements of the transaction that third parties want to authenticate. Then, the signing party encrypts the result with their private key.

11.2. How blockchains work

A blockchain is a set of individual transactions grouped into blocks, where each block contains the transactions emitted since the last block was added to the blockchain. Each transaction is emitted by a member node that has already been enrolled, which then broadcasts it to all the members of the blockchain.

The authenticity[5] and legitimacy[6] of the transaction are then verified by the other nodes of the blockchain, which rely on the history of transactions recorded since the beginning of the blockchain. Then, miners combine all approved transactions into a batch that they add to the block they are building. They validate the block by mining, i.e. solving a complex math-

4 In the context of this document and of cryptography in general, "very difficult" suggests that current algorithms and computing resources cannot allow for an attack on a hash function in a reasonable time frame (several billiard years on one computer).

5 An *authentic* transaction is a transaction for which the emitting node has been authenticated.

6 A *legitimate* transaction is a transaction that the emitting node is authorised to initiate (i.e. the node has sufficient funds).

ematical puzzle called Proof of Work (PoW).[7] The first miner to solve the mathematical puzzle broadcasts the solution to all the nodes, which check the PoW. Once the solution is approved and the block has been added to the blockchain, miners begin to mine for the next block. As many nodes contribute to writing the block into the blockchain, this process relies on a consensus among nodes — this consensus principle becomes an essential characteristic of the governance structure of the blockchain.

Different types of nodes

From a technical perspective, members of the blockchain are computing resources (i.e. computers) that are connected to the blockchain through an *enrolment* phase. They belong to a network connected through the Internet and are usually called *nodes*.

To become a member of a blockchain, a person therefore needs to enrol a computer resource as a node. There are two types of nodes::

- ***regular nodes***, which for the most part have regular computing power, from which transaction requests can be emitted;
- ***miner nodes***, with large computing power that is useful to the blockchain, also able to submit transactions.

Both types of nodes can store the whole blockchain, provided they have enough memory. They are then called *full nodes*. The Bitcoin blockchain, launched in 2009, was more than 190GB in 2018.

The enrolment phase

During enrolment, nodes, both regular and miner, download a software that enables them to interface with the blockchain. This software is tailored to a personal blockchain account number (i.e. a 160-bit Bitcoin address) and a set of public and private keys. The node owner is required to keep the software and password to access their private key. If they lose one or the other, access to the blockchain account will be lost and no transaction may ever be emitted from that account again.

The link between the account number and the public key needs to be obvious and easy to check in order to authenticate the origin of a transaction request. In the case of the

7 The Proof of Stake scheme is fundamentally different from the Proof of Work, as explained page 190.

Bitcoin, the address is simply the result of the hash function on the public key, so that any node can authenticate the owner of an account as the entity behind a transaction. This bypasses the need for a key management infrastructure, which is interesting because managing electronic certificates[8] is both burdensome and costly.

The transaction phase

In the transaction phase, all transactions are validated, combined into a block, then mined (through PoW or PoS) — which typically takes several minutes (around 10 minutes for the Bitcoin project). The new block is then broadcast and added to the blockchain, after checking that the mining was successful.

Each blockchain gives the initiating node a certain degree of freedom regarding the conditions that need to be met for the transaction to be legitimate and authentic.

For Bitcoin, the implicit legitimacy condition is that a node should possess more Bitcoins than it is trying to transfer. The initiating node may also add a script requirement for authentication conditions: for instance, that the beneficiary node prove its identity by sending a valid digital signature, or multiple ones in case the owner owns multiple accounts and wishes to augment the level of security.

For Ethereum, the conditions are set by Smart Contract authors.

Regardless of the specific conditions adopted by each blockchain, a transaction always needs to contain (see Figure 8):

- a unique transaction identifier.
- information enabling to verify the transaction and to the least establish its context. In the bitcoin blockchain, it is required to provide inputs to a transaction that enable the initiating node (Bertrand) to identify anterior transactions (Anne's and Alice's) and check the legitimacy and authenticity of the current one: whether Bertrand has the necessary Bitcoin resources as well as the cryptographic conditions that are required by Anne and Alice (i.e. a public key and a digital signature) to prove he is the recipient of their money transfer.

8 An electronic certificate is a data structure that links a public key with its owner's ID in a secure way.

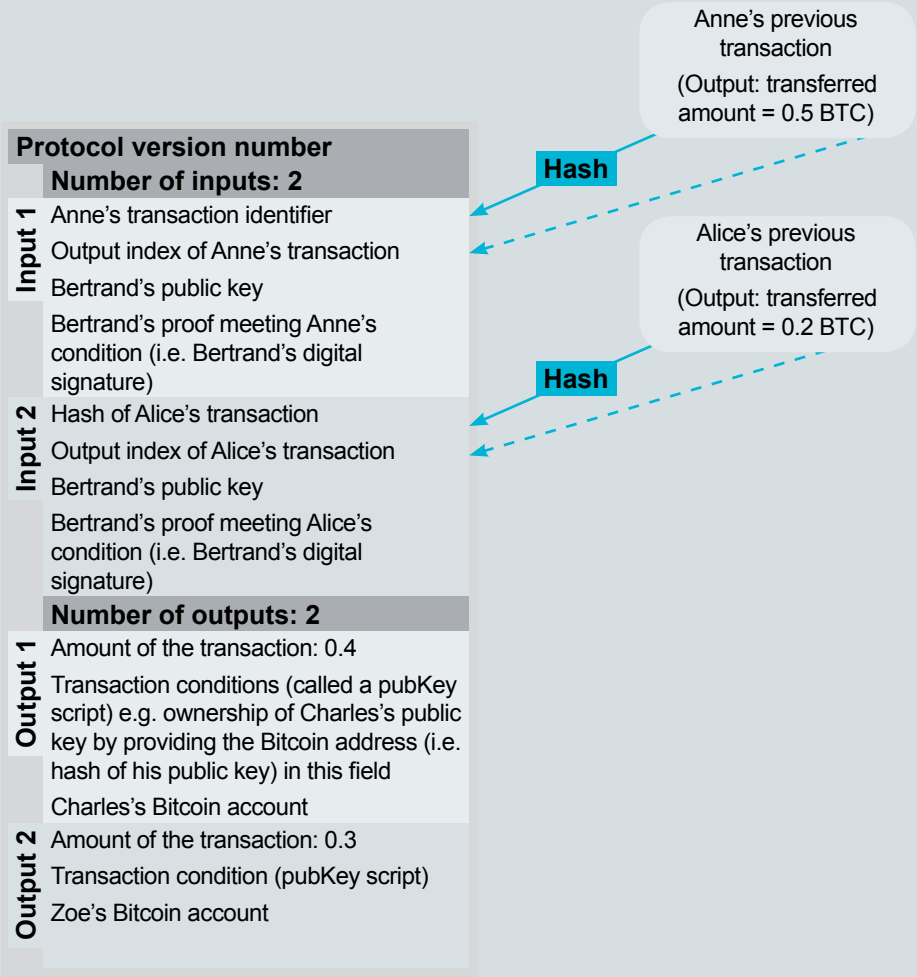

Figure 8. Simplified structure of a Bitcoin transaction

- information on the transaction result. Bitcoin also specifies outputs such as the transaction's recipients (Charles and Zoe), the amount, and the conditions that recipients need to meet to claim this sum. As in any ledger, inputs and outputs can have equal amounts, and if an output is lower than the sum of the inputs, then the miner receives the difference to compensate for the mining work. Such transaction fees are sometimes necessary to incentivise miners to prioritise transactions offering higher compensations in the block they are mining, leading to miners competing for the highest-paying transactions on the blockchain.

Creating the blocks

A block is made up of a batch of transactions, which it writes into a block so that their content as well as the position of the block within the blockchain cannot be altered in the future, be it through an accident or an attack. This protection against accidents and attacks relies on two necessary complementary processes.

The first process provides the series of transactions and blocks with a chained structure by linking them into a chain. This process relies heavily on hash functions and on the principle of a Merkle tree.[9] Hash functions prevent the partial modification of a block within the blockchain, which would trigger the avalanche effect, but they cannot protect against overwriting the last blocks, as we explain in section 11.3. For these blocks, the mechanism of mining together with a decentralised storage and computing architecture offers a level of trust. The elements providing a structural as well as a functional measure of trust are presented in 11.3.

As regards the specific Bitcoin structure (see Figure 9), a block contains a header including technical information on the blockchain, content including transactions, and a nonce, which is a random number used for mining, as well as other elements we explain below.

During each transaction, an identifier is computed (TxID), equal to the hash of the transaction's content. The Merkle tree then enables to securely add this transaction to the chain by calculating the hashes of all the blocks up to the root of the tree. The result of these

9 "*A Merkle tree is a tree in which every leaf node is labelled with the hash of a data block and every non-leaf node is labelled with the cryptographic hash of the labels of its child nodes. [Merkle] trees allow efficient and secure verification of the contents of large data structures.*" Wikipedia: https://en.wikipedia.org/wiki/Merkle_tree

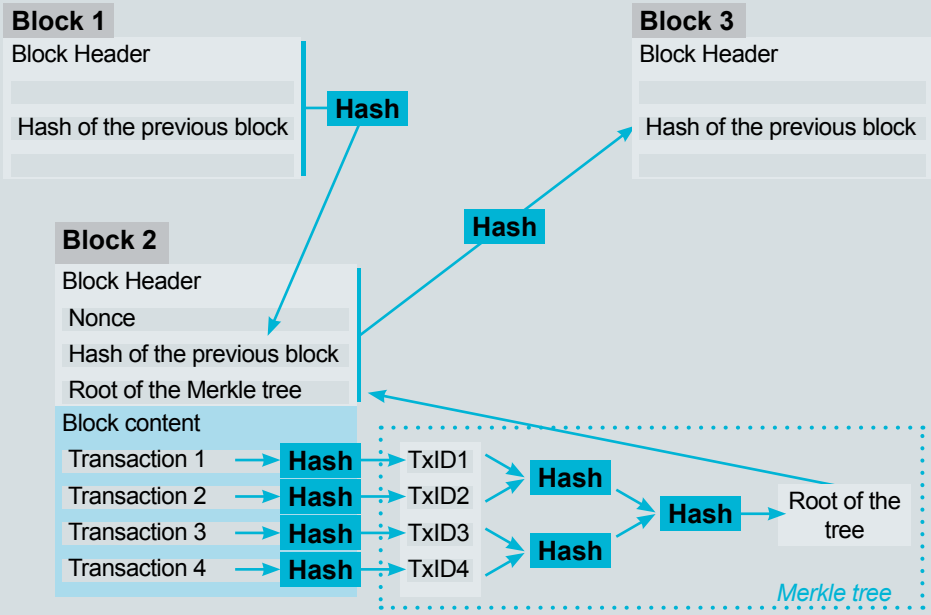

Figure 9. Simplified format of a Bitcoin block and its chaining to the blockchain

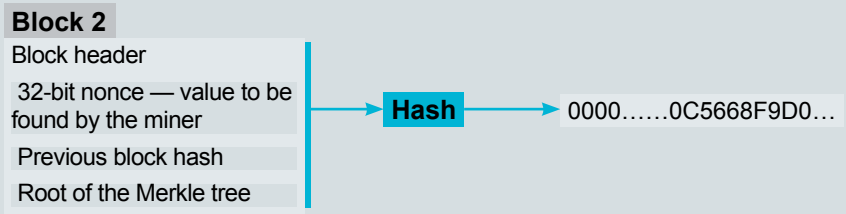

Figure 10. PoW mining on Bitcoin header

calculations is then written in the block's header, which is one way to check the block's integrity later on.[10]

The second process guarantees the integrity of the block's place within the blockchain by chaining the blocks into a series starting with the "Genesis Block."

► In Figure 9, block 2 is chained between blocks 1 and 3. Its location can be verified by checking that block 1's hash corresponds to the hash in block 2's header, and similarly block 2's hash in block 3's header.

When the PoW puzzle is solved by two (or more) miners simultaneously, the other nodes receive two different validated blocks. These are both added to the chain at the same level, which in practice creates a fork and two different blockchains.

This forking problem is self-regulated, because the mining effort is unequally distributed between the two temporarily distinct blockchains. The blockchain relying on the largest amount of computing power will grow faster and thus be recognised as valid — this is a first security vulnerability, as we explain in more detail in 11.3. For the Bitcoin blockchain, once 100 blocks have been added, forking problems are supposed to be solved. This obviously implies adding the transactions of the abandoned blockchain that are not in the validated blockchain back into it. Another convention is that a Bitcoin transaction is only considered effective once it's been buried under 6 blocks, which requires an hour wait before the recipient can use the Bitcoins it has received for another transaction. This condition is one of the main issues of blockchains in dynamic environment, which has led researchers to consider alternatives such as Proof of Stake (see 11.2).

How mining operations are confirmed

Mining enables miners to build a valid block by solving a complex mathematical puzzle and earning compensation for it. Before solving the puzzle, the miner adds a transaction named "coinbase" in the block for this compensation. The blockchain policy consists in enabling the miner to create some amount of currency and to ignore the rule for standard

10 To check the integrity of a block, the verifier has to successively perform the exact same Merkle tree hashing operations to locally compute the Root of the Merkle tree, then has to check that the result matches the value given in the field "Root of the Merkle Tree" of the block header. In case it does not match, the block is detected as corrupted.

transactions according to which the output needs to be lower than the input. The miner thus uses the "coinbase" transaction to specify the recipient and amount of the reward. This way, after the puzzle is solved and if the block is accepted by the other miners, this amount and the transaction fees are transferred to the miner who solved the puzzle. For Bitcoin, the reward per block decreases over time, and miners increasingly rely on transaction fees for their compensation.

Validation by Proof of Work (PoW)

To validate a block, miners have to solve a puzzle, i.e. to find a 32-bit nonce to add to the block's header so that the header's hash is lower than a threshold value called *difficulty* (see Figure 10). The lower this threshold, the harder the problem. The blockchain policy is to adjust the difficulty so that the difficulty (interpreted as a level of security) remains constant.

- For Bitcoin, a block is validated every 10 minutes, on average. After 2016 blocks have been validated, which takes around two weeks, the average time is calculated. If it is too short, the difficulty is increased; if too long, it is decreased.

One of the major issues with PoW is that miners are required to use a lot of computing power. To solve this issue, the Proof of Stake (PoS) process was developed.

Validation by Proof of Stake (PoS)

Validation through PoS is a simpler process than validation through PoW. It enables to both reduce energy demand and make the blockchain more dynamic. This affects the sustainability and economic incentives, as the blockchain will then be able to record transactions more quickly and thus handle a larger volume of transactions.

- The Ethereum project is currently developing a PoS algorithm called Casper. Migration towards Casper should start in 2018 with a hybrid PoS-PoW version, and progressively replace PoW. PoS is expected to speed up the validation of blocks up to more than 20,000 transactions per second.

From a practical perspective, PoS validation is even more decentralised than PoW. Indeed, PoW requires nodes to compete on the same puzzle, creating redundancy in how the computing power is allocated, from the validation of the last transaction to the

moment the puzzle solution is found. Because computing power is unequally distributed, some nodes have more influence than others on the outcome of collective decisions on the blockchain, all the more since much of the computing power used to mine Bitcoins is located in China. The Casper process functions differently: it does not distribute transactions amongst all nodes but divides them into subgroups. The system then favours nodes with the highest engagement, e.g. those with the most Bitcoins, which means that they have the most to lose in case of malicious behaviour. Further, a system of fines exists to punish negative behaviour.

While the PoS system is promising in theory, we do recommend caution: it is currently being tested in Ethereum but is nowhere near as reliable as PoW, which has already withstood large-scale experiments in Bitcoin and Ethereum.

Mining incentives

Mining is an essential part of the blockchain. A gain, or "crypto fuel", that is valued enough is therefore needed to incentivise miners to contribute computing resources and to store the blockchain locally. This compensation needs to offset the economic costs of computing material (required material has very high computing power and/or very high storage capacities), its maintenance as well as energy costs.

As a reminder, miners need a lot of computing power. A compensation is paid for each successful mining operation that is accepted by the peers and added into a block (see "How mining operations are confirmed" in 11.2.).

Blockchain designers define what kind of "crypto fuel" will be produced. It is usually related to the blockchain's activity — bitcoin for the Bitcoin blockchain, ether for the Ethereum blockchain — but can also be designed as part of a loyalty programme: free storage space, computing power, voting power, a car rental, a hotel stay or a trip.

Whichever "crypto fuel" is chosen, the incentive requires a virtual unit that enables miners to accumulate gains depending on how much effort they put in, as a classic loyalty card would do.

11.3. Trust factors

A blockchain consists of several features that can induce trust — however not complete trust.

Decentralised architecture and governance neutrality

Firstly, trust relies on a ***decentralised architecture***, with a large number of nodes belonging to different organisations. Unlike in a centralised architecture where decisions can be taken without consensus, one needs to either produce some level of consensus or control more than 50% of the nodes (or the computing power) to act on the system as a whole. Since the architecture relies on many nodes, the work of validating and storing transactions in the blockchain, as well as any updates to the rules governing the blockchain, need to receive consensus from a broad group of stakeholders, thus forbidding a small group to become too influential in the governance mechanisms.

Trust requires computing resources and storage capacities to be balanced among organisations; yet we observe the exact opposite situation in the Bitcoin blockchain, with the creation of mining pools. The largest three pools have held more than 50% of the network's computing power on several occasions already. This 50% threshold is critical because it enables an organisation or a coalition of organisations to implement a 51% attack: essentially, to be able to control the history of transactions, but not necessarily to steal currency gains nor add malicious transactions.[11]

Secondly, trust relies on a ***neutral governance scheme*** — the blockchain equivalent to the notion of balance of powers. Before investing time and money into a blockchain, one needs to check whether the neutrality of the governance scheme is guaranteed: whether the limited number of people managing the project and its protocol are really independent in their decision-making process and resistant to political or industrial pressure. If such is not the case, then power in the blockchain is fundamentally not balanced. Further, if these stakeholders control more than half of the computing power, the consensus principle does not hold either. Indeed, when the blockchain operating rules are updated through an up-

11 A 51% attack is an attack on the blockchain that filters transactions before the mining process and directs the gains of the mining efforts to its own miners instead of those who are the fastest. In the case of competing blockchains, a group holding more than 50% of the mining power could theoretically allocate their mining power to one of the competing blockchains and therefore decide on the issue of the conflict with confidence.

date of the blockchain's code, miners and their administrators may either accept or reject the update. This can be a minor and backward-compatible update — called a *soft fork* — or a major and not backward-compatible update — called a *hard fork*. To be implemented, a soft fork only requires the support of a majority of miners, whereas a hard fork requires a much larger consensus. In the event a large consensus is not obtained but large-enough groups support both solutions, the blockchain divides into two different blockchains that survive on their own. Therefore, a coalition of stakeholders who hold most of the mining capacity could collude, modify the governance rules, create forks and confusion, create double spending (see below), and risk devaluating the cryptocurrency as a whole.

Transparency enables better auditability

Trust also relies on ***transparency***. This principle applies at many levels, including transactions and algorithms.

- **Traceability and auditability of the entire chain of transactions:** The publication of all transactions recorded from the Genesis Block enables all nodes to verify the integrity of the chain and obtain all the transactions associated with an account. In theory, fraud is therefore impossible: all is public and transparent, in the limits provided by pseudonymity.

- **Algorithmic transparency:** Anybody can read the code used for mining, interacting with the blockchain and implementing a smart contract. This gives experts among the user community the opportunity to scrutinise the code and raise a red flag if they notice anything suspicious. Trust therefore largely relies on watchdogs.

Digital security

Finally, blockchains enable good digital risk management (see Chapter 4) through three main features:

- **A rigid tamper-proof chain:** Both the content of the blocks within the blockchain and their order are tamper-proof. This relies on the decentralised architecture and the consensus principle. On top of this, there can be a mechanism incentivising positive behaviour, disincentivising negative behaviour, and a cryptographic system supporting strong technical guarantees. The PoW relies on consensus and a cryptographic proof that is costly in terms of computing power, while the PoS relies

on consensus and an incentive structure and has not yet proven it could be trusted at a large scale.

- **The ability to authenticate transactions while protecting digital identities:** Blockchains provide privacy (e.g. through the use of pseudonyms) yet implement adapted security measures to guarantee that transactions are valid and that accounts are secure. This balance between identity protection and security management is a crucial factor in trusting the blockchain.

- **Security levels can be tailored:** As new technologies are developed, security mechanisms once deemed trustworthy become vulnerable. To maintain the same level of trust, several blockchains enable security levels to be dynamic.

However, trust in the blockchain can never be complete. Several elements have actually questioned this trust, following these events:

- **Programming errors:** Programmable blockchains imply a high risk of human programming errors, as happened with the 2016 attack on Ethereum. In 4 weeks, the Decentralised Autonomous Organisation (the DAO),[12] which enables its community to invest in venture capital, raised a spectacular amount of $150 million to fuel start-up projects wishing to build over Ethereum. The DAO was then robbed of $50 million by a group of hackers who exploited a vulnerability in the way smart contracts were implemented. This error enabled the attackers to use the function designed to "cash out" an account several times. As Ethereum co-founder Vitalik Buterin wrote in a blog post, "*This is an issue that affects the DAO specifically; Ethereum itself is perfectly safe.*"[13] In 2017, another attack on the wallet software Parity Wallet led to $30 million in ether being stolen..

- **Double spending:** The double spending problem arises when one single piece of currency is used in two different transactions, which should normally exclude each other. This is a voluntary and malicious act, which the mining process deletes under

12 Blockchain France defines a DAO as "*an organisation that relies on a computer software to define rules governing the community. These rules are transparent and immutable, as they are written into the blockchain.*" [Unofficial translation from the French]

13 https://blog.ethereum.org/2016/06/17/critical-update-re-dao-vulnerability/

normal conditions. It can however happen that each mutually exclusive transaction is recorded on a forked chain. In this case, the recipient can only figure out whether they received the transaction once one of the two blockchains is abandoned. For Bitcoin, a reasonable timeframe is 1 hour, i.e. 6 blocks later. The problem of double spending was one of the major issues with online currencies before Bitcoin offered a practical solution to it: the blockchain.

- **Appropriating transactions:** It may be in a miner's interest not to share a transaction with a high fee to other miners. By mining the transaction by himself, the miner ensures they will be the one to receive the transaction fee — but it might take more time for the transaction to get included in the blockchain. This retention attack is becoming more likely as transaction fees are increasing while built-in rewards are decreasing. Similarly, a well-connected miner may choose to retain a block to get more time to mine and broadcast it broadly only when he has received a competitor's block. This type of attacks questions the incentive system and calls for improvements..

- **Money laundering:** Money laundering issues appear every time a new way of exchanging money is created. Contrary to popular belief, transaction transparency does not prevent money laundering; it only makes it more complex. Indeed, some techniques can be used to decrease traceability. Firstly, one can create a multiplicity of accounts (some only used once) and a network of transactions between those accounts. A second approach, called Coinjoin and used in Bitcoin, consists in combining several transactions into a single one. The more transactions are merged (inputs and outputs), the harder it is to link a spender to a recipient. The Zerocash approach we describe in 11.4. guarantees that transactions are non-traceable and makes it impossible to detect money laundering on the sole basis of information acquired from the blockchain.

11.4. Transparency and privacy breaches in the blockchain

A blockchain relies on the pseudonymity of its participants, which means that once the real identity of an account holder is revealed, all of the transactions they made from their account can be revealed. As explained above, many techniques can protect users' real identity, including owning multiple accounts (some only used once) and merging transactions, as is possible with Coinjoin.

The transparency of the blockchain should cause service designers to be more cautious as to the protection of personal data. Indeed, any private information, be it algorithms or data (e.g. personal data, cryptographic keys…), should not be stored unencrypted in the blockchain, for instance in a transaction. However, since it is in any case better to limit the size of the information stored in the blockchain to limit costs, one may still rely on distributed storage systems. Such systems can rely on an externalised, potentially distributed and unlimited memory: they can be implemented to function as a peer-to-peer network[14] (e.g. BitTorrent, GNutella, Napster or Kademlia). In this case, the memory is actually externalised because the content is accessible through a Distributed Hash Table (DHT) key and only this key needs to be referenced in the blockchain.[15] This memory can then store either encrypted or unencrypted data — in the case of encrypted data, there is then a need to manage cryptographic keys.

In 2014, the Zerocash initiative offered an interesting solution for decentralised anonymised payments.[16] This solution enables transparent and untraceable Bitcoin transfers on a blockchain: neither the source, the destination, nor the amount can be inferred. The solution relies on zero-knowledge protocols (where neither party reveals information to the other) that enable a user to prove to a third party they know a secret without having to reveal the secret itself. This relies on zero-knowledge Succinct Non-interactive ARguments

14 A peer-to-peer (P2P) network is a network built over the Internet and made of P2P nodes assigning a portion of their resources for the P2P service, mostly file sharing application, to be provided to the community with the idea that peers are equally privileged and powerful in the application.

15 A DHT key associated to a content can be easily computed by applying a hashing function over the content. This key needs to be known in order to access the associated content stored in a P2P network. To go into detail, the participating P2P nodes share in a distributed way a DHT table including for each entry a DHT key (itself assigned to a content) and a value useful for peers to locate the P2P node where the content is stored. Note that any node is able to compute that value by hashing the DHT key.

16 Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., Virza, M., (2014). Zerocash: Decentralized Anonymous Payments from Bitcoin, 2014 IEEE Symposium on Security and Privacy.

of Knowledge (zk-SNARKs), particularly efficient since they are able to establish a proof of knowledge in a handful of milliseconds. To explain how this works, the following image is often used: all users pin their banknotes on a wall and remove them when they make a transaction.

Finally, in 2015, the MIT developed a solution called Enigma, which offers a decentralised cloud platform ensuring the confidentiality of all the data processed and the computing operations.[17] It relies on blockchain to ensure the traceability of operations and on the Enigma peer-to-peer network to compute and store sensitive data. The idea is that each Enigma node only possesses an incomplete and meaningless view of the sensitive data being processed, and only processes it partly. Therefore, nodes cannot individually access sensitive information. Through Secure Multi-Party Computing (SMC), they can collaboratively produce the result sought by the system.

11.5. What are the current limits of the blockchain?

We have seen that blockchain technologies have structural limits. They cannot be considered as a basis for complete trust and confidence, even narrowed down to trust. Indeed, organisational issues relating to power dynamics between actors and user appropriation as well as technical factors make studying the actual scope of this technology very complex. However, they indicate once more that mere transparency does not necessarily come with complete trust and an adequate protection of personal data.

Let us finally remind here that Public Key Infrastructures (PKI) were once similarly presented as a revolutionary, trust-inducing technology, before we came to share an understanding of its limits.

Therefore, and as is the case with labels in a broader sense, using a blockchain is a guarantee of certain properties, but should be considered as a way to induce or suggest user trust by emphasizing the appropriate features of this technology.

17 Zyskind, G., Nathan, O., Pentland, A., (2015). Enigma: Enigma: Decentralized Computation Platform with Guaranteed Privacy, http://enigma.media.mit.edu/enigma_full.pdf

How to cite this chapter: Laurent M. “Is blockchain a trustworthy technology?”, *in Signs of trust – The impact of seals on personal data management*, Paris, Handbook 2 Chair Values and Policies of Personal Information, Coordinated by Claire Levallois-Barth, January, 2018, chapter 11, pages 179–197.

http://www.personal-information.org/